

ACCESS ALKALMAZÁS KIALAKÍTÁSA DOLGOZÓI JELENLÉT NYILVÁNTARTÁSÁRA

Fabulya Zoltán

Absztrakt: Egyedi, testreszabott program szolgáltatásait tudjuk biztosítani az Access adatbázis-kezelő program segítségével. Felhasználóbarát kezelőfelületekkel rendelkező alkalmazásokat fejleszthetünk programozói ismeretek nélkül. Áttekintjük egy portán hasznosítható, dolgozók mozgását nyilvántartó adatbázis fejlesztését: az adatok tárolását biztosító relációs adatbázis tábláinak és kapcsolatainak kialakítása, a szükséges lekérdezések, űrlapok, jelentések és makrók elkészítése. A programszerű működést biztosító beállításokkal bárki számára könnyen használható felületeket alakíthatunk ki a program egyszerű kezeléséhez.

Abstract: We can provide the services of a customized program with the help of Microsoft Access database management system. Applications with user-friendly interfaces can be developed without programming knowledge. We give a review of the development of a database which can be used at a reception to track the movement of employees: the creation of the tables and their relationships in the relational database to store data, the required queries, forms, reports and macros. We can create an easy-to-use user interface with the appropriate settings which make the database act like a program.

Kulcsszavak: Microsoft Access, adatbázis, alkalmazás beállításai, kapcsolótábla

Keywords: Microsoft Access, database, configuring application, switchboard

1. Bevezetés

Az Access adatbázis-kezelő program szolgáltatásaival egyszerűen tudunk kialakítani olyan adatkezelő alkalmazást, mely egy egyedi fejlesztésű, testreszabott program képességeivel rendelkezik. Ennek ellenére mégsem elterjedt a használata. Ez elsősorban azzal magyarázható, hogy kevesen ismerik azt az elméleti hátteret, mely szükséges egy relációs adatbázis megtervezéséhez. Ehhez fel kell mérnünk a tárolandó adatok esetén a köztük lévő kapcsolatokat, funkcionális függéseket. Az így adódó adattároló táblák rendszerének még ahhoz is igazodnia kell, hogy milyen adatkezelési funkciókat akarunk megvalósítani, mennyire kényelmes, felhasználóbarát módon (Bártfai, 2004).

A fejlesztendő alkalmazás egy korábbi kiegészítése, mellyel egy portán nyilvántarthatjuk a dolgozók mozgását, aktuális tartózkodási helyét. Ehhez ki kell alakítani az adatok tárolásához a szükséges táblákat és a köztük lévő kapcsolatokat. Megtervezzük az adatbeviteli felületeket űrlapokkal, majd az aktuális igényeknek megfelelő adatokat visszakereső lekérdezéseket, s az azokat megjelenítő jelentéseket. Végül kialakítjuk a funkciók elérését kényelmesen biztosító menüt.

2. Anyag és módszer

A személyes adatok védelme érdekében csak a tervek, a kialakított funkciók, az alkalmazás beállításai és működése kerülnek bemutatásra.

Az adatbázisban tárolni szeretnénk minden alkalmazott esetén:

- a nevét,
- munkahelyre érkezésük és távozásuk pontos időpontját dátummal együtt,
- napközbeni mozgásukat
 - o távozási és érkezési időpontját,
 - o a távozás célját, jellegét (hivatalos, magán).

Ezek az adatok teszik lehetővé, hogy fontos jelenléti információkat kapjunk nem csak a pillanatnyi állapotról, de így visszamenőleg is összesített eredményeket képezhetünk.

Az adatbázis kialakításához és használatához az Access 2010 programot alkalmazzuk, mely igen széleskörűen alkalmazható, akár a pénzügy (Zsótér–Túri, 2017), vagy a biometria területén (Hampel, 2017). A tervező fázisban kell elkészítenünk az adatbázis összes olyan objektumát és az ezekhez szükséges kezelő felületet, melyek biztosítják egy laikus számára is az egyszerű és jól áttekinthető használatot. Arról is gondoskodni kell, hogy csak az adatokhoz legyen hozzáférése az objektumainkon keresztül a felhasználónak, míg a tervek megváltoztatása kellő védelemmel rendelkezzen (Fabulya, 2018).

Az adatbázis megfelelő működtetését különböző objektum típusok biztosítják, melyek a következők:

- tábla,
- lekérdezés,
- űrlap,
- jelentés,
- makró.

A táblák az adatok tárolására alkalmasak. A táblákban olyan mezőnek is lennie kell, mely a tábla bármely rekordjának egyértelmű azonosítására képes, azaz elsődleges kulcsként alkalmazható. A táblák között lévő kapcsolatok kialakításakor az elsődleges kulcs mező szükséges.

Amikor adatokat szeretnénk kigyűjteni egyetlen táblázatosan értelmezhető eredménnyé, ehhez választó lekérdezéseket biztosít az Access program. Feltételek megadásával szűrhetjük a kapott rekordokat, s még számított és összesítő eredményeket is képezhetünk (Demeter, 2000).

Felhasználóbarát adatkarbantartás kezelői felületét űrlapok többféle nézetben teszik lehetővé. Adatlap nézetben táblázatos formát kapunk, míg űrlap nézetben csak egy sor (rekord) adatait szerkeszthetjük.

Jelentésekkel nyomtatáshoz szükséges elrendezést képezhetünk, csoportosító és összesítő beállításokat is megadhatunk. Sokszor nem a nyomtatás miatt, csak az eredmények monitoron megjelenítéséhez alkalmazzuk (Halassy, 1995).

Makrók alkalmasak speciális funkciók ellátására. Parancsok sorozatát tudjuk egy makró segítségével elvégezni (Czenky, 2007).

Ezek az objektumok határozzák meg az adatbázis szolgáltatásait. Ezek felhasználóbarát menü jellegű elérését kapcsolótáblával valósíthatjuk meg. Az adatbázis indítási beállításával megoldhatjuk, hogy az adatbázis megnyitásakor ez a

kapcsolótábla jelenjen meg, s még az objektumok tervének védelmére is itt nyílik mód.

3. Eredmények és értékelésük

3.1. Az adatrögzítés és adatkarbantartás eszközei, a táblák és űrlapjaik kialakítása

Az 1. ábra mutatja az adatok tárolására kialakított táblák rendszerét. A dobozok egy-egy táblát szimbolizálnak, bennük a mezőkkel, míg a köztük lévő vonalak a kapcsolatot jelentik az elsődleges kulcs szerepét betöltő mező segítségével. Megtervezéséhez fel kellett mérni a használathoz szükséges funkciókat, adatokat, s még a kényelmes kezelhetőséget is, mint például a legördülő listás adatbeviteli technika alkalmazhatósága olyan esetekben, ahol így kiváltható a gépelés.

1. ábra: Az adatbázis táblái és kapcsolatuk



Forrás: A szerző saját szerkesztése.

A *Jelenlét* és a *Napközbeni távozások* táblák a két központi objektum. Ezek adatai folyamatosan bővülnek az adatbázis használata során. A *Dolgozó* és *Távozási Cél* táblák adatai ritkán változnak. Szerepük, hogy a jelenléti adatok kezelése közben listás kiválasztást kínáljanak gépelés helyett.

Egy tábla tervezésekor be kell állítani:

- a mezőket,
- a mezők adattípusát,
- az elsődleges kulcsot,
- az adatbevitel és megjelenítés módját.

A 2. ábrán a *Jelenlét* tábla kialakításának beállításait láthatjuk. Az elsődleges kulcs szerepét betöltő *Jelenlétazon* mező számláló típusú, ami azt jelenti, hogy sorszám jellegűen automatikusan keletkezik egy következő érték minden újabb érkező dolgozó regisztrálásakor, akinek a neve helyett egy öt azonosító szám tárolódik egy kombinált listában megjelenő dolgozói névsorból választás hatására,

de a dolgozó neve kerül megjelenítésre. Az *Érkezés* és *Távozás* mezők megfelelő adattípussal tárolják az időpontokat.

2. ábra: A *Jelenlét* tábla tervező nézete

Jelenlét		
	Mezőnév	Adattípus
	Jelenlétazon	Számláló
	Dolgozó	Szám
	Érkezés	Dátum/Idő
	Távozás	Dátum/Idő
Mezőtulajdonságok		
Általános	Megjelenítés	
Vezérlőelem megjelenése	Kombinált lista	
Sorforrás típusa	Tábla/lekérdezés	
Sorforrás	SELECT Dolgozó.Név, Dolgozó.Az FROM Dolgozó ORDER BY Dolgozó.Név;	
Kötött oszlop	2	
Oszlopszám	1	
Oszlopfejlécek	Nem	

Forrás: A szerző saját szerkesztése.

A *Napközi távozások* tábla hasonló adatokat tárol, mint a *Jelenlét* tábla azzal a kiegészítéssel, hogy a távozás célja is rögzíthető listából választással.

Az adatbázis használata során egy dolgozó érkezésekor, illetve távozásakor újabb adatok rögzítését a *Jelenlét* táblába a 3. ábrán látható *Munkába érkezés távozás* űrlappal tehetjük meg. Itt nem jelenik meg a *Jelenlétazon* mező, hiszen ez automatikusan keletkezik minden rekord esetén, s csak zavaró lenne a felhasználó számára. Új rekord csak érkezéskor keletkezik, amikor meg kell adni listából választással a dolgozó nevét és az érkezés időpontját, míg távozáskor csak ennek időpontja rögzítendő.

3. ábra: A *Jelenlét* tábla adatkezelő űrlapja

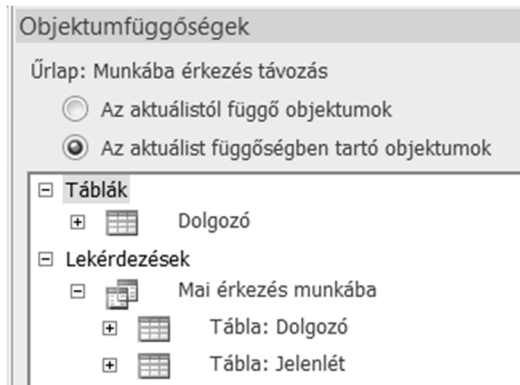
Munkába érkezés távozás		
Dolgozó	Érkezés	Távozás
* <input type="text"/>	<input type="text"/>	<input type="text"/>

Forrás: A szerző saját szerkesztése.

Viszont azt is meg kell oldani, hogy távozáskor az űrlap kínálja fel azt a rekordot, ahol a dolgozó érkezése már rögzítésre került. Mindeközben mégse jelenjen meg a *Jelenlét* tábla összes rekordja. Ennek megoldásaként az Űrlap alapja nem lehet a

tábla, hanem annak csak azon rekordjai, melyek mai napi érkezéssel alakultak ki. E feltételnek megfelelő rekordokat egy lekérdezéssel alakíthatjuk ki az űrlap számára. Így az adatok rögzítését a táblába az űrlap nem közvetlenül, hanem a lekérdezés segítségével áttételesen teszi lehetővé. Ennek megfelelően objektum függőségek jelennek meg az adatbázisban, melyet a *Munkába érkezés távozás* űrlap esetén a 4. ábrán láthatjuk.

4. ábra: A *Munkába érkezés távozás* űrlap objektumfüggősége

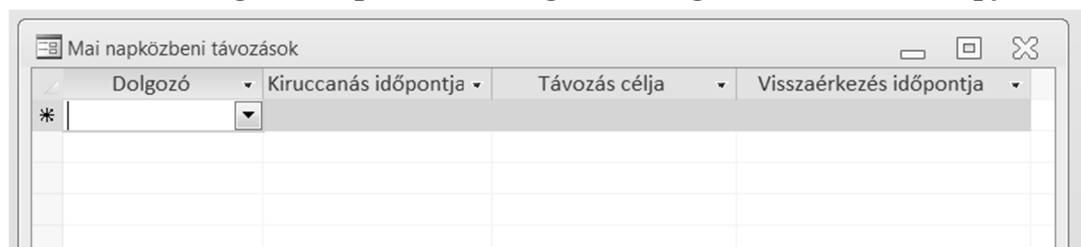


Forrás: A szerző saját szerkesztése.

A 4. ábra alapján a *Munkába érkezés távozás* űrlap közvetlen alapja a *Dolgozó* tábla és a *Mai érkezés munkába* lekérdezés. Az utóbbi pedig a *Dolgozó* és *Jelenlét* táblákon alapul. Ez a lekérdezés tartalmazza a mai napi érkezés feltételét.

A Napközbeni távozások adatait szintén egy felhasználóbarát felületet nyújtó űrlappal rögzíthetjük (5. ábra). A *Dolgozó* és *Távozás* célja mezők értéke legördülő listás választással adható meg. Itt sem közvetlen a kapcsolat a táblával, mert csak a mai naphoz tartozó rekordok érdekesek, melyeket egy lekérdezés feltétele eredményez.

5. ábra: *Dolgozók napközbeni mozgásakor rögzítendő adatok* űrlapja



Forrás: A szerző saját szerkesztése.

A *Dolgozó* és *Távozási Cél* táblák adatainak kényelmes kezelése egy-egy űrlappal közvetlenül történik.

3.2. Adatkereső, összesítő funkciók

Két összesítő szolgáltatással rendelkezik a program, Az egyik a dolgozók pillanatnyi jelenlét eredményezi a monitoron jól áttekinthető formában, a másik a tárolt jelenléti adatok visszakeresését teszi lehetővé egy adott év vagy hónap esetén, melyet nyomtatható formában gyűjt ki.

A pillanatnyi jelenlét megjelenítését az Aktuális jelenlét űrlap biztosítja (6. ábra). A szükséges adatok kigyűjtése a táblákból több fázisban, egymáson alapuló lekérdezések segítségével valósul meg a probléma összetettsége miatt. A 7. ábrán látható ennek objektumfüggősége. Egyszerre elvárt cél, hogy minden szükséges információ megjelenjen, de mégis egyszerűen áttekinthető formában. Az űrlap adatlapos szerkezetét és írásvédettségét makró biztosítja.

6. ábra: Az Aktuális jelenlét űrlap

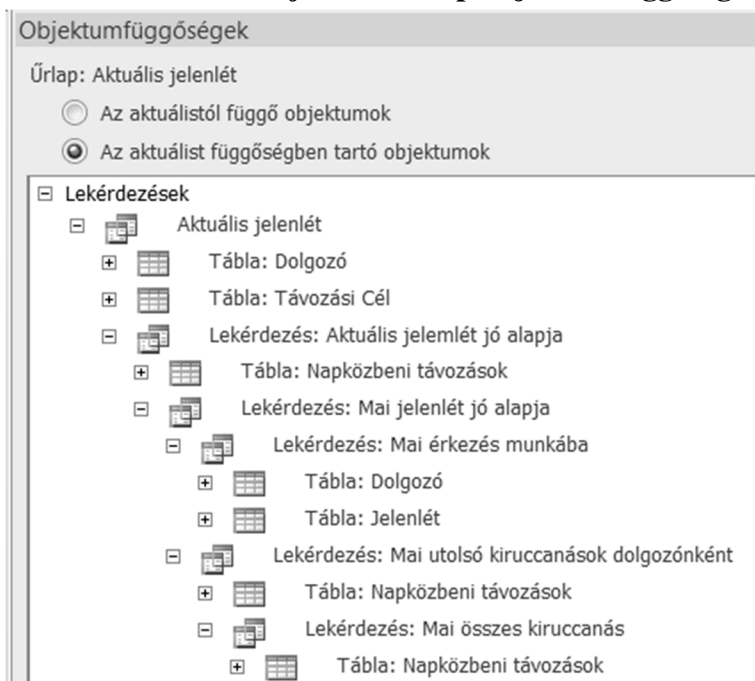
Név	Érkezés	Távozás	Kiruccanás idő	Távozási cél	Visszaérkezés időp

Forrás: A szerző saját szerkesztése.

Ahhoz, hogy a dolgozók pillanatnyi jelenlét megkapjuk, a következőket kellett feltételekkel megoldani:

- Minden dolgozó jelenjen meg, de csak egyszer, függetlenül attól is, hogy beérkezett-e, vagy esetleg már befejezte a munkát.
- Ha még nem érkezett be a munkába, akkor csak a neve jelenjen meg.
- Ha már beérkezett, akkor több napközbeni távozása esetén is csak az utolsót mutassa.
- Ha még nem tért vissza napközbeni távozásból, jelezze a távozás időpontját.
- Ha már befejezte a munkát, akkor ezt a távozás időpontjával jelezze napközbeni távozások adatai nélkül.

7. ábra: Az Aktuális jelenlét űrlap objektumfüggősége



Forrás: A szerző saját szerkesztése.

Az utolsó funkciója a programnak a dolgozók jelenléti adatainak visszakeresését szolgálja. Ezt egy évre, vagy egy hónapra kapjuk meg nyomtatható formában. Az eredmény egy részletét láthatjuk a 8. ábrán, ahol a név adatai itt elrejtésre került.

8. ábra: A Dolgozók jelenléti adatai jelentés részlete

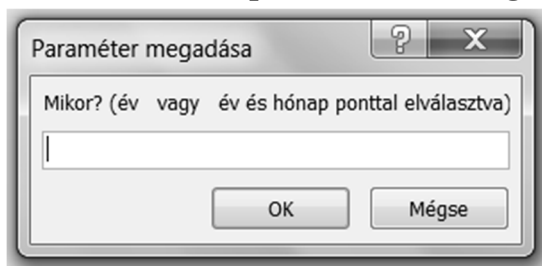
Dolgozók jelenléti adatai						
Név	Érkezés	Távozás	Távozási cél	Kiruccanás	Visszaérkezés	Hivatalos
	2011. 6. 24. 8:57	17:06				<input type="checkbox"/>
	2011. 6. 27. 7:57	14:48	Humán	10:25	11:31	<input checked="" type="checkbox"/>
			Egyéb	12:45	14:22	<input type="checkbox"/>
	2011. 6. 28. 8:02	14:00	Bolt	8:17	8:19	<input type="checkbox"/>
			Humán	8:23	8:55	<input checked="" type="checkbox"/>
	2011. 6. 29. 8:14	14:14				<input type="checkbox"/>

Forrás: A szerző saját szerkesztése.

A Dolgozók jelenlétéről rögzített minden adat megjelenik dolgozónként csoportosítva, a csoporton belül időrendi sorrendben. Meg kellett oldani, hogy a funkció végrehajtásakor rákérdezzen a program a kívánt időszakra. Ez egy paraméteres lekérdezéssel valósítható meg, ahol a felhasználó által begépelt adatot veszi figyelembe a program az adatok szűréséhez. Az időszak megadását a 9. ábra

mutatja, míg az ehhez szükséges beállítások a lekérdezés tervének kialakításakor a 10. ábrán láthatók.

9. ábra: A lekérdezés paraméterének megadása



Forrás: A szerző saját szerkesztése.

10. ábra: Paraméteres lekérdezés tervének részlete

Mező:	Érkezés
Tábla:	Jelenlét
Rendezés:	Növekvő
Megjelenítés:	<input checked="" type="checkbox"/>
Feltétel:	Like Left([Mikor? (év vagy év és hónap ponttal elválasztva)];7) & ""
vagy:	

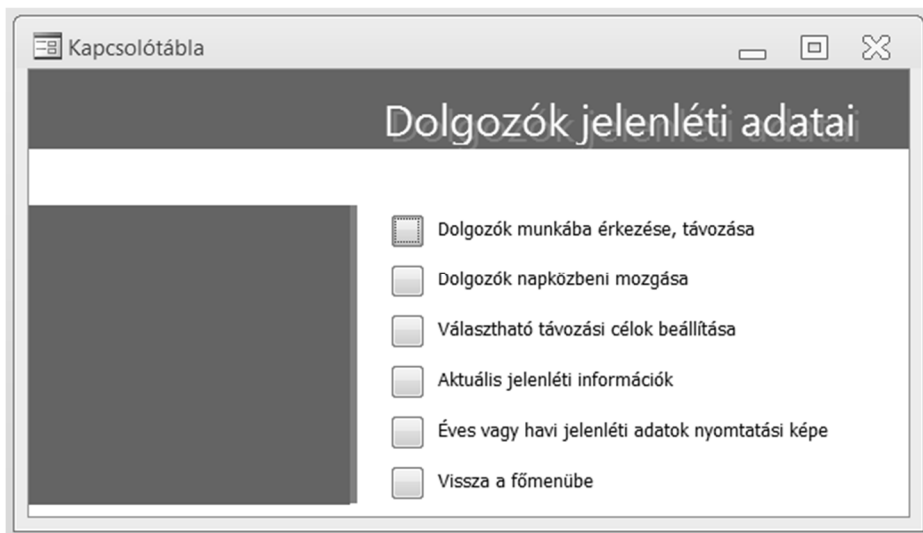
Forrás: A szerző saját szerkesztése.

A lekérdezés tervezőrácsában feltételként adunk meg szögletes zárójelek között egy kérdést, s az erre begépelt válasz helyettesítődik be ennek helyére, mint a paraméter aktuális értéke. Tehát így más-más feltételekkel képes eredményeket adni a lekérdezés. Az ábrában az látható, hogy a Jelenlét tábla érkezés mezőjében tárolt adatra annak a feltételnek kell teljesülnie, hogy a paraméterként begépelt szövegnek legfeljebb első 7 karakterében megegyezzen.

3.3. Kapcsolótábla, avagy menü készítése a funkciókhoz

Az Access adatbázis-kezelő program *Kapcsolótábla* nevű űrlap kialakításával támogatja a kialakított funkciók elérhetőségét menüként használható módon. Mivel egy korábbi program kiegészítését végeztük el, így ez a menü már elkészült, csak ki kellett egészíteni egy almenüvel (11. ábra).

11. ábra: A dolgozók jelenléti adatainak kapcsolótáblája

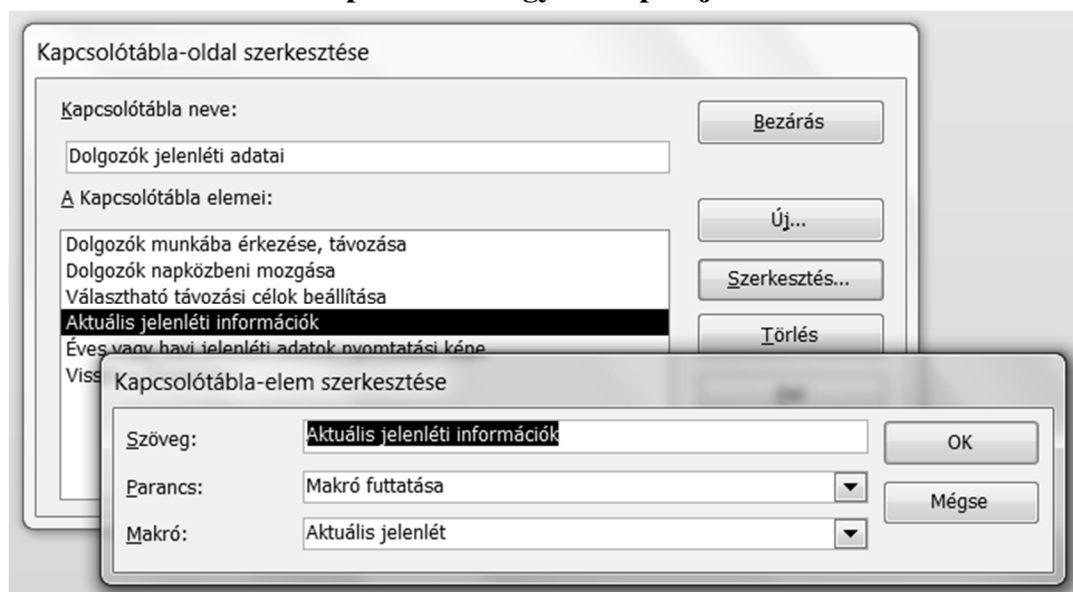


Forrás: A szerző saját szerkesztése.

Az első három menüpont az adatbeviteli funkciókhoz kell, a következő kettő az adatok visszakereséséhez, míg az utolsóval a főmenübe jutunk vissza.

Egy kapcsolótábla elkészítéséhez az Access „kapcsolótábla kezelő” szolgáltatását kell használnunk (12. ábra). Az ábra egy menüpont kialakítását mutatja. Ehhez elegendő megadnunk a megjelenítendő szöveget, majd legördülő listából kiválaszthatjuk a szükséges parancsot, s annak objektumát. Az *Aktuális jelenlét* nevű makró futtatása történik a menüpont kiválasztásakor.

12. ábra: A kapcsolótábla egy menüpontjának kialakítása



Forrás: A szerző saját szerkesztése.

Az adatbázis védelmét úgy kell kialakítani, hogy a felhasználó ne tudjon a terveken módosítani, és csak a számára kialakított funkciókat érje el. Ez az adatbázis beállításával lehetséges. Le kell tiltani az objektumok szabad elérhetőségét, valamint a menüt biztosító kapcsolótábla űrlapjának megnyitását kell beállítani az adatbázis megnyitásakor.

4. Következtetések, összegzés

Egyszerű eszközökkel ki tudunk alakítani olyan felhasználói felületet az Access adatbázis-kezelő programmal, mellyel saját készítésű menün keresztül tehetjük elérhetővé a kialakított funkciókat. Így programként viselkedő, könnyen kezelhető alkalmazást készíthetünk egy átlagos számítógép használó számára.

Irodalomjegyzék

- Bártfai B. (2004): *Adatbázis-kezelés*. BBS-Info, Budapest.
- Czenky M. (2007): *Access programozás: 2003-as és 2007-es verzió*. ComputerBooks, Budapest.
- Demeter M. I. (2000): *Egyszerűen Access 2000 for Windows*. Panem, Budapest.
- Fabulya Z. (2018): Access alkalmazás kialakítása ügyfélközpontú szolgáltatások nyilvántartására. *Jelenkori társadalmi és gazdasági folyamatok*, 13 (1): 67–76.
- Halassy B. (1995): *Az adatbázis-tervezés alapjai és titkai*. IDG Magyarországi Lapkiadó Kft., Budapest.
- Hampel Gy. (2017): Excel VBA alkalmazása egy biometria esettanulmány példáján bemutatva. *Jelenkori társadalmi és gazdasági folyamatok*, 12 (4): 34–40.
- Zsótér B., Túri I. (2017): Economical calculations related to a smoking technology investment of a pork processing plant. *Annals of Faculty of Engineering Hunedoara – International Journal Of Engineering*, 15 (4): 57–61.